

BEST PRACTICE SERVICE-ORIENTED ARCHITECTURE (SOA): RECOMMENDATION AND TIPS

Maria Seraphina Astriani

Computer Science Department, School of Computer Science Binus University
Jl. K.H. Syahdan No. 9, Palmerah, Jakarta Barat 11480
seraphina.astriani@gmail.com

ABSTRACT

Companies that have many applications always deal with application integration problems. For the of company necessity, they often yearn for information that an application can be used by other applications. Often a small portion of data from another application is needed in order to complete and support other business functions. They require an architectural approach that can bridge the information between applications. SOA (Service-Oriented Architecture) offers manufacturing services around existing business functions. Applications that wish to communicate with other applications can use the services to achieve business goals. This article will discuss recommendations and tips for implementing SOA best practices in banking.

Keywords: *application integration, SOA, best practices*

ABSTRAK

Perusahaan yang mempunyai banyak aplikasi selalu dirundung masalah pengintegrasian aplikasi. Untuk keperluan perusahaan, seringkali mereka menginginkan informasi satu aplikasi untuk dapat digunakan oleh aplikasi lainnya. Seringkali sebagian kecil data dari aplikasi lain sangat dibutuhkan agar dapat menyelesaikan dan mendukung fungsi bisnis yang lainnya. Mereka membutuhkan suatu pendekatan arsitektur yang dapat menjembatani informasi antar aplikasi. SOA (Service-Oriented Architecture) menawarkan pembuatan services disekitar fungsi bisnis yang telah ada. Aplikasi yang ingin berkomunikasi dengan aplikasi lainnya dapat menggunakan services untuk mencapai tujuan bisnis. Penulisan ini akan membahas rekomendasi dan tips pengimplementasian SOA secara best practice di dalam perbankan.

Kata kunci: *integrasi aplikasi, SOA, best practices*

PENDAHULUAN

Aplikasi bisnis tidak dapat berdiri sendiri. Setiap aplikasi haruslah dapat berkomunikasi dengan aplikasi-aplikasi lainnya agar dapat saling bertukar informasi dan dapat saling mendukung informasi antara satu dengan yang lainnya. Bahkan sebagian kecil data dan semantik dari sistem yang lain sangat dibutuhkan agar dapat menyelesaikan dan mendukung fungsi bisnis.

Service-oriented architecture (SOA) merupakan *new face* dari *enterprise application integration* (EAI). SOA dapat membantu membuat desain menjadi lebih baik, “lincah”, dan dapat digunakan kembali, menghemat biaya untuk *maintenance*, serta telah menjadi pilihan untuk pendekatan EAI (Seshadri, 2007).

Perusahaan yang mempunyai banyak aplikasi selalu dirundung oleh permasalahan EAI. Untuk keperluan perusahaan, seringkali mereka menginginkan informasi satu aplikasi dapat digunakan oleh aplikasi lainnya. Bahkan sebagian kecil data dari aplikasi lain sangat dibutuhkan agar dapat menyelesaikan dan mendukung fungsi bisnis.

Perusahaan, sebagai contoh bank, sudah menjadi bagian di dalam awal dunia bisnis yang mengadopsi dan menggunakan bantuan komputerisasi karena penggunaan komputasi mekanik dapat membantu memberikan keuntungan dalam transaksi keuangan. Bahkan tidaklah aneh jika pada saat ini masih banyak ditemukan aplikasi-aplikasi bank yang menggunakan bahasa pemrograman lama yaitu C dan Basic karena program tersebut sudah digunakan sejak lama dan sangat sulit jika aplikasi tersebut dimigrasi dengan menggunakan bahasa lebih baru.

Dengan bertambahnya aplikasi, suatu ”bahasa” komunikasi yang sama diperlukan agar dapat saling berbicara dengan aplikasi lainnya. Untuk hal tersebut diperlukan SOA untuk dapat membantu perusahaan menjalankan bisnisnya.

SOA bukanlah sebuah teknologi, melainkan sebuah pendekatan arsitektur yang dibangun disekitar teknologi yang sudah ada. SOA menawarkan suatu *practices, disciplines, design*, dan *guidelines* yang dapat di aplikasikan pada sebuah atau banyak teknologi (Seshadri, 2007).

SOA menawarkan pembuatan *services* di sekitar fungsi bisnis yang telah ada. Aplikasi yang ingin berkomunikasi dengan aplikasi lainnya dapat menggunakan *services* untuk mencapai tujuan/tugas bisnis (Seshadri, 2007).

Prinsip dari SOA adalah membangun suatu rancangan antar muka standar untuk dapat mengakses fungsi-fungsi bisnis yang berlainan, yang ada pada berbagai macam sistem aplikasi utama suatu perusahaan (Seshadri, 2007).

Kasus diambil dari bank, suatu jenis perusahaan yang sangat menggantungkan bisnisnya kepada sistem. Aplikasi yang digunakan di dalam bank sangat banyak dan akan terus bermunculan aplikasi lainnya yang akan dipakai. Pada saat ini bank memperluas bisnisnya bukan hanya sebagai tempat untuk menabung, tetapi juga menawarkan *credit card*, asuransi, cicilan, dan lainnya. Oleh karena itu penambahan aplikasi bank pasti akan bertambah jumlahnya.

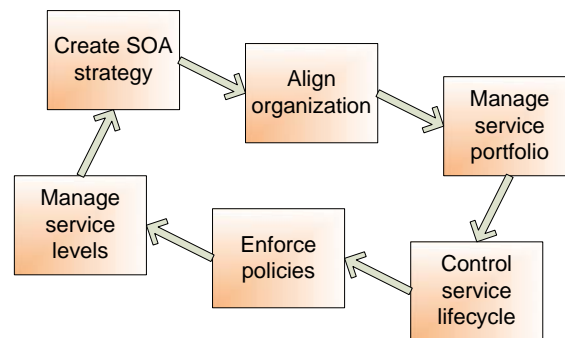
Semua aplikasi diatas membutuhkan suatu media komunikasi besar yang dapat digunakan untuk berbagai sistem. Jika permasalahan tidak diatasi, akan berakhir pada komunikasi yang sulit untuk dimengerti dan dikelola. Jika permasalahan ini dibiarkan berlanjut, akan menyebabkan informasi bisnis tidak dapat diperoleh oleh semua sistem karena tidak saling terintegrasi. Hal ini dapat mendatangkan kerugian yang cukup besar pada perusahaan seperti hilangnya kesempatan bisnis,

berkurangnya kepuasan pelanggan, dan bahkan dapat menciptakan permasalahan-permasalahan baru lainnya. Pada saat ini, tantangan yang ada untuk setiap perusahaan dan institusi keuangan adalah bagaimana caranya mengadopsi dan menerapkan strategi arsitektur di dalam sistem agar dapat sejalan dengan visi bisnis dan mengatasi masalah komunikasi antar aplikasi. Solusi dari tantangan ini akan dijawab pada sub bab Hasil dan Pembahasan.

METODE

Metode yang digunakan pada studi implementasi SOA secara best practices ini merupakan tinjauan kepustakaan yang diambil dari beberapa sumber.

Siklus SOA dimulai dari membuat strategi SOA. Strategi SOA dibangun sesuai dengan kebutuhan bisnis yang ada. Sebagai pengukur bahwa pembangunan SOA berjalan dengan baik, diperlukan suatu *Quality of Service* (QoS) yang terdapat Service Level Agreement untuk setiap *service* yang dilakukan. Jika telah sampai mengelola *service level*, siklus dapat kembali ke siklus awal. Metode SOA terangkum pada Gambar 1 berikut.



Gambar 1. Metodologi SOA (Schepers, 2008).

HASIL DAN PEMBAHASAN

Rekomendasi dan Implementasi SOA

SOA *best practices* yang mengindikasikan struktur *request* dan *response* haruslah terinspirasi/berasal dari fungsi bisnis dan model data bisnis. *Data management team* disarankan untuk turut serta pada awal mendesain spesifikasi *enterprise service* untuk mencapai tingkat efektifitas lebih baik di dalam pekerjaan ini.

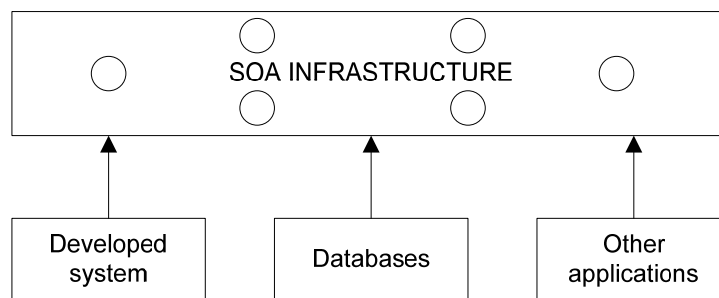
Platform SOA haruslah fleksibel sehingga dapat mendukung *client* dengan *fixed, delimited*, dan berbagai macam format data, serta dapat memperbolehkan *services* mengakses ke berbagai macam protokol (Seshadri, 2007). *Services* harus sekurang-kurangnya dapat mendukung *messaging* atau *transport protocol*: JMS (Java Message Service), HTTP(S), TCP/IP (Internet Protocol), dan MQ. Protokol-protokol mempunyai berbagai macam metodologi untuk *load balancing* dan *scaling*. Normal metodologi diharuskan mempunyai *layer* yang dapat mentransformasikan semantik antara non-XML dan format XML.

SOA tidak dapat berjalan sukses jika tidak digunakan pada berbagai aplikasi bisnis yang ada pada perusahaan. Berikut ini merupakan cara-cara yang direkomendasi agar dapat menjalankan SOA dengan baik (Seshadri, 2007). Pertama, komunikasi merupakan kunci untuk keberhasilan. Dengan menghadiri seminar, mengadopsi teknologi baru yang baik dapat membantu dalam memahami dan mengadopsi SOA dengan lebih baik. Kedua, melibatkan orang-orang yang mengerti tentang SOA dari bagian yang berbeda di dalam perusahaan, untuk dapat membantu mengintegrasikan kebutuhan-kebutuhan divisi perusahaan secara menyeluruh agar dapat sesuai dengan *services* yang dibutuhkan. Ketiga, tim haruslah secara proaktif memimpin proyek agar dapat berjalan dengan baik dan memastikan integrasi penggunaan SOA telah sesuai dengan kebutuhan.

Pada saat penerapan *reuseability* ‘kegunaan kembali’ perlu diperhatikan agar *services* dapat bekerja pada bisnis yang berlainan (Bieberstein et al, 2008). Poin-poin yang ada di dalam *reuseability* yaitu: (1) pilih *services* yang dapat dipakai dalam jangka waktu yang lama; (2) pilih *service* karena tidak semua *services* dapat digunakan kembali; (3) *services* harus dirancang secara spesifik. Umumnya diperlukan infrastruktur perangkat keras yang baru untuk mengakomodasi SOA. Untuk membuat infrastruktur SOA diperlukan orang yang mengatur kegiatan tersebut. *SOA operation center* dapat dibentuk; (4) Seperti kebanyakan *operation center*, *SOA operation center* dapat terdiri dari bagian: *development*, System Integration Test, User Acceptance Test, serta *disaster recovery*. Kapasitas dari perangkat keras yang ada di dalam *platform* SOA harus diperhatikan dari waktu ke waktu, dan perangkat keras tambahan dapat di tambahkan ke dalamnya jika dibutuhkan.

Tips pengimplementasian SOA adalah sebagai berikut: (1) adopsi pembuatan IT Blueprint, perhatikan elemen perancangan SOA (Astriani et al, 2010), di antaranya: *people*, *business process*, *information flow*, dan *tools / system*; (2) sebisa mungkin jangan ubah *interface* karena dapat menyebabkan ketidaksinkronan dalam *parsing parameter* (Lero et al, 2007); (3) pastikan ada pengecekan dan pengetestan di setiap fase. *Test cases* dapat diutilisasi dan digunakan kembali untuk aplikasi baru lainnya (Ziang et al, 2004).

Gambar 2 dibawah ini merupakan gambaran arsitektur dari penerapan SOA di dalam perbankan. Lingkaran menandakan *services* yang dibungkus di dalam infratruktur SOA. *Developed system* merupakan sistem utama bank, sedangkan *other applications* merupakan aplikasi-aplikasi pendukung lainnya seperti asuransi, cicilan, dan sebagainya.



Gambar 2. Infrastruktur SOA (Semenchuk, n.d.)

PENUTUP

SOA merupakan pendekatan arsitektur yang memodularisasikan sistem informasi ke dalam *services*. Stabilitas *interface* merupakan kunci kesuksesan SOA. SOA dapat membuat desain menjadi lebih baik, “lincah”, dan dapat digunakan kembali, serta menghemat biaya untuk *maintenance*.

Jika di dalam sistem terdapat banyak aplikasi yang tidak dapat berintegrasi, diperlukan SOA sebagai pembungkus agar aplikasi dapat berkomunikasi dengan yang lainnya. Di dalam segi bisnis, SOA mempunyai suatu nilai tambah karena dapat memberikan informasi strategis yang dapat membantu perkembangan bisnis dan *executive decision*. Rekomendasi beserta tips penerapan SOA dapat membantu sebagai pedoman dalam membangun infrastruktur SOA.

DAFTAR PUSTAKA

- Astriani, M. S., Pradono, S., Saragih, H. (2010). *IT Blueprint for Education Institution*. Advances in Computing, Control and Telecommunication Technologies (ACT), 2010 Second International Conference. IEEE Xplore.
- Bieberstein, N., Laird, R. G., Jones, K., Mitra, T. (2008). *Executing SOA: A Practical Guide for the Service-Oriented Architect*. IBM Press.
- Lero, L. O., Merson, P., Bass, L. (2007). *Quality Attributes for Service-Oriented Architectures*. International Workshop on System Development in SOA Environment (SDSOA'07). IEEE.
- Schepers, T.G.J., Iacob, M.E., dan Van Eck, P.A.T. (2008). *A Lifecycle Approach to SOA Governance*. SAC'08, March 16-20, 2008. ACM.
- Semenchuk, Pavel. *Improvement of Applications Development Using Service Oriented Architecture*. Riga Technical University.
- Seshadri, Gokul. (2007). *Case Study Implementing SOA: Methodology and Best Practices*. IGI Global: USA.
- Ziang, J., Chung, Jen-Yao., Chang, C. K. (2004). *Migration to Web Services Oriented Architecture – A Case Study*. 2004 ACM Symposium on Applied Computing.